# INTERNAL MODEL CONTROL (IMC) AND IMC BASED PID CONTROLLER

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

**Bachelor of Technology in**

**Electronics and Instrumentation Engineering**


**By**

**ANKIT PORWAL**

**ROLL NUMBER: 10607001**

**&**

**VIPIN VYAS**

**ROLL NUMBER: 10607009**

**Department of Electronics & Communication Engineering**

**National Institute of Technology**

**Rourkela**


**2009-2010**

# INTERNAL MODEL CONTROL (IMC) AND IMC BASED PID CONTROLLER

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

**Bachelor of Technology in**

**Electronics and Instrumentation Engineering**

**Under the Guidance of**

***Prof. T K DAN***

**BY**

**ANKIT PORWAL (10607001)**

**&**

**VIPIN VYAS (10607009)**

**Department of Electronics & Communication Engineering**

**National Institute of Technology**

**Rourkela**

**2009-2010**

## NATIONAL INSTITUTE OF TECHNOLOGY
## ROURKELA

## CERTIFICATE

This is to certify that the project report titled **"INTERNAL MODEL CONTROL (IMC) AND IMC BASED PID CONTROLLER "** submitted by Ankit Porwal (Roll No: 10607001) and Vipin Vyas ( Roll No: 10607009)in the partial fulfillment of the requirements for the award of Bachelor of Technology Degree in Electronics and Instrumentation Engineering during session 2006-2010 at National Institute of Technology, Rourkela (Deemed University) and is an authentic work carried out by them under my supervision and guidance.

To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other university/institute for the award of any Degree or Diploma.

**Prof. T K DAN**

Date:

**Department of E.C.E**

**National Institute of Technology**

**Rourkela-769008**

# ACKNOWLEDGEMENT

We would like to take this opportunity to express my gratitude and sincere thanks to our respected supervisor **Prof. T K DAN** for his guidance, insight, and support he has provided throughout the course of this work.

We are also grateful to our respected **Prof. U C PATI**. Under the guidance of respected professors we learned about the great role of self-learning and the constant drive for understanding emerging technologies, and a passion for knowledge.

We would like to thank all faculty members and staff of the Department of Electronics and Communication Engineering, N.I.T. Rourkela for their extreme help throughout course.

**ANKIT PORWAL (10607001)**

**VIPIN VYAS (10607009)**

# CONTENTS

**SIMULATION RESULTS**

# ABSTRACT

**Internal Model Control** (IMC) is a commonly used technique that provides a transparent mode for the design and tuning of various types of control. The ability of proportional-integral (PI) and proportional-integral-derivative (PID) controllers to meet most of the control objectives has led to their widespread acceptance in the control industry. The Internal Model Control (IMC)-based approach for controller design is one of them using IMC and its equivalent IMC based PID to be used in control applications in industries. It is because, for practical applications or an actual process in industries PID controller algorithm is simple and robust to handle the model inaccuracies and hence using IMC-PID tuning method a clear trade-off between closed-loop performance and robustness to model inaccuracies is achieved with a single tuning parameter.

Also the IMC-PID controller allows good set-point tracking but sulky disturbance response especially for the process with a small time-delay/time-constant ratio. But, for many process control applications, disturbance rejection for the unstable processes is much more important than set point tracking. Hence, controller design that emphasizes disturbance rejection rather than set point tracking is an important design problem that has to be taken into consideration.

In this thesis, we propose an optimum IMC filter to design an IMC-PID controller for better set-point tracking of unstable processes. The proposed controller works for different values of the filter tuning parameters to achieve the desired response As the IMC approach is based on pole zero cancellation, methods which comprise IMC design principles result in a good set point responses. However, the IMC results in a long settling time for the load disturbances for lag dominant processes which are not desirable in the control industry.

In our study we have taken several transfer functions for the model of the actual process or plant as we have exactly little or no knowledge of the actual process which incorporates within it the effect of model uncertainties and disturbances entering into the process. Also, the parameters of the physical system vary with operating conditions and time and hence, it is essential to design a control system that shows robust performance in the case of the above mentioned situations. Then we tried to tune our IMC controller for different values of the filter tuning factor.

Since all the IMC-PID approaches involve some kind of model reduction techniques to convert the IMC controller to the PID controller so approximation error usually occurs. This error becomes severe for the process with time delay. For this we have taken some transfer functions with significant time delay or with non invertible portions i.e. containing RHP poles or the zeroes. Here we have used different techniques like factorization to get rid off these error containing stuffs. It is because if these errors are not removed then even if IMC filter gives best IMC performance but structurally causes a major error in conversion to the PID controller, then the resulting PID controller could have poor control performance.

Thus in our approach to IMC and IMC based PID controller to be used in industrial process control applications, there exists the optimum filter structure for each specific process model to give the best PID performance. For a given filter structure, as $\lambda$ decreases, the inconsistency between the ideal and the PID controller increases while the nominal IMC performance improves. It indicates that an optimum $\lambda$ value also exist which compromises these two effects to give the best performance. Thus what we mean by the best filter structure is the filter that gives the best PID performance for the optimum $\lambda$ value.

## List of Figures



## List of tables

# Chapter 1

## INTRODUCTION TO
## INTERNAL MODEL CONTROL (IMC)

## 1.1 IMC Background

In process control applications, model based control systems are often used to track set points and reject low disturbances. The internal model control (IMC) philosophy relies on the internal model principle which states that if any control system contains within it, implicitly or explicitly, some representation of the process to be controlled then a perfect control is easily achieved. In particular, if the control scheme has been developed based on the exact model of the process then perfect control is theoretically possible.

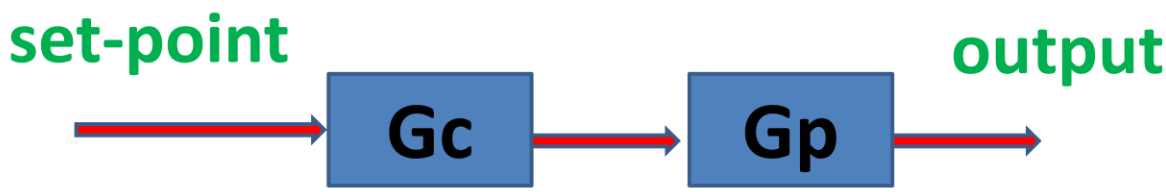



Fig 1. 1 Open loop control strategy

For above open loop control system:

**Output = Gc . Gp . Set-point (multiplication of all three parameters)**

**Gc = controller of process**

**Gp = actual process or plant**

**Gp* = model of the actual process or plant**

A controller Gc is used to control the process Gp. Suppose Gp* is the model of Gp then by setting:

**Gc =inverse of Gp*** (inverse of model of the actual process)

And if

**Gp = Gp*** (the model is the exact representation of the actual process)

Now it is clear that *for these two conditions the output will always be equal to the set point.*

It show that if we have complete knowledge about the process (as encapsulated in the process model) being controlled, we can achieve perfect control.

This ideal control performance is achieved without feedback which signifies that feedback control is necessary only when knowledge about the process is inaccurate or incomplete.

Although the IMC design procedure is identical to the open loop control design procedure, the implementation of IMC results in a feedback system. Thus, IMC is able to compensate for disturbances and model uncertainty while open loop control is not. Also IMC must be detuned to assure stability if there is model uncertainty.

## 1.2 IMC basic structure

The distinguishing characteristic of IMC structure is the incorporation of the process model which is in parallel with the actual process or the plant. Also we consider that '*' is generally used to represent signals associated with the model.

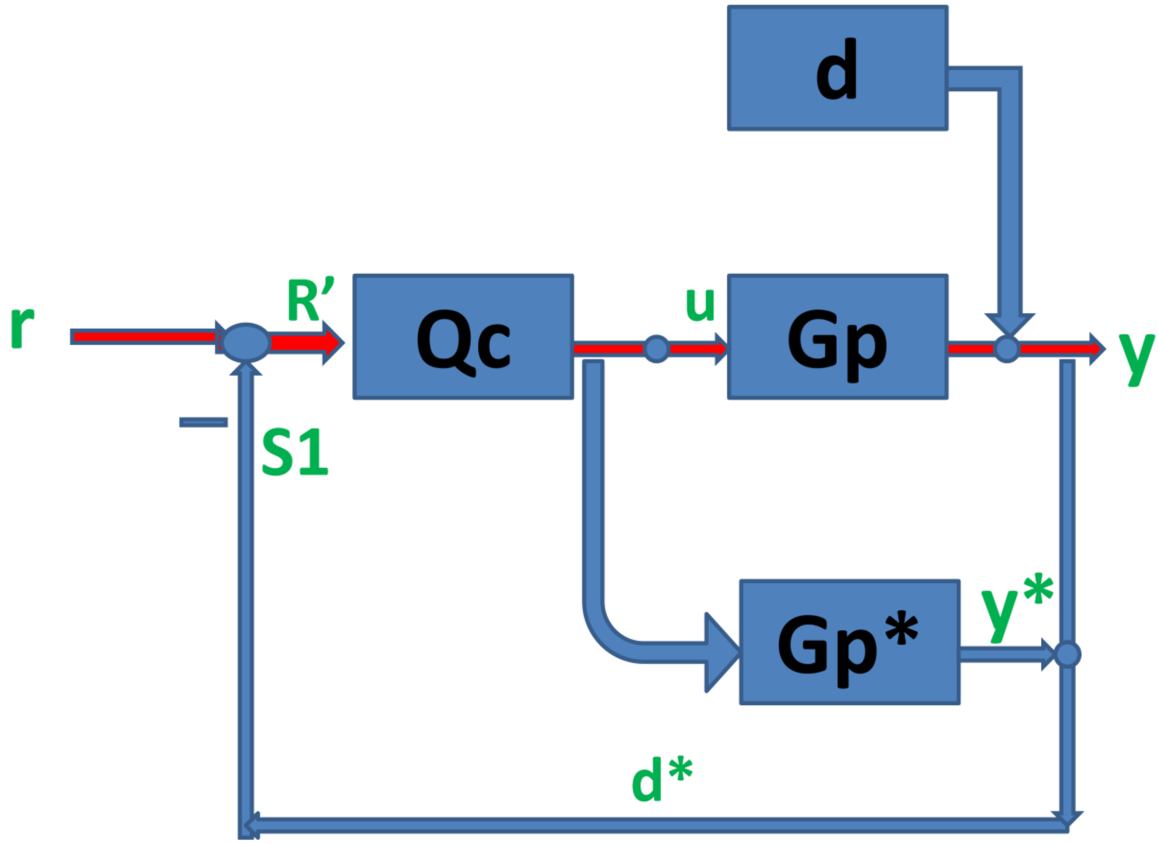



**Fig 1. 2 IMC basic structure**

### 1.3 IMC parameters

The various parameters used in the IMC basic structure shown above are as follows:

Qc= IMC controller

Gp= actual process or plant

Gp*= process or plant model

r= set point

R'= modified set point (corrects for model error and disturbances)

u= manipulated input (controller output)

d= disturbance

d*= estimated disturbance

y= measured process output

y*= process model output

Feedback signal:

d*= (Gp - Gp*)u +d

Signal to the controller:

R'= r- d*= r- (Gp - Gp*) u – d

Now we consider a limiting case

**Perfect model with no disturbance:**

We will say a model to be perfect if

Process model is same as actual process

i.e. **Gp = Gp***

no disturbance means

**d = 0**

Thus we get a relationship between the set point **r** and the output **y** as

**y = Gp . Qc .r**

This relationship is same for as we got for open loop system design. Thus if the controller Qc is stable and the process Gp is stable the closed loop system will be stable.

But in practical cases always the disturbances and the uncertainties do exist hence actual process or plant is always different from the model of the process.

## 1.4 IMC Strategy

As stated above that that actual process differs from the model of the process i.e. process model mismatch is common due to unknown disturbances entering into the system. Due to which open loop control system is difficult to implement so we require a control strategy through which we can achieve a perfect control. Thus the control strategy which we shall apply to achieve perfect control is known as INTERNAL MODEL CONTROL (IMC) strategy.

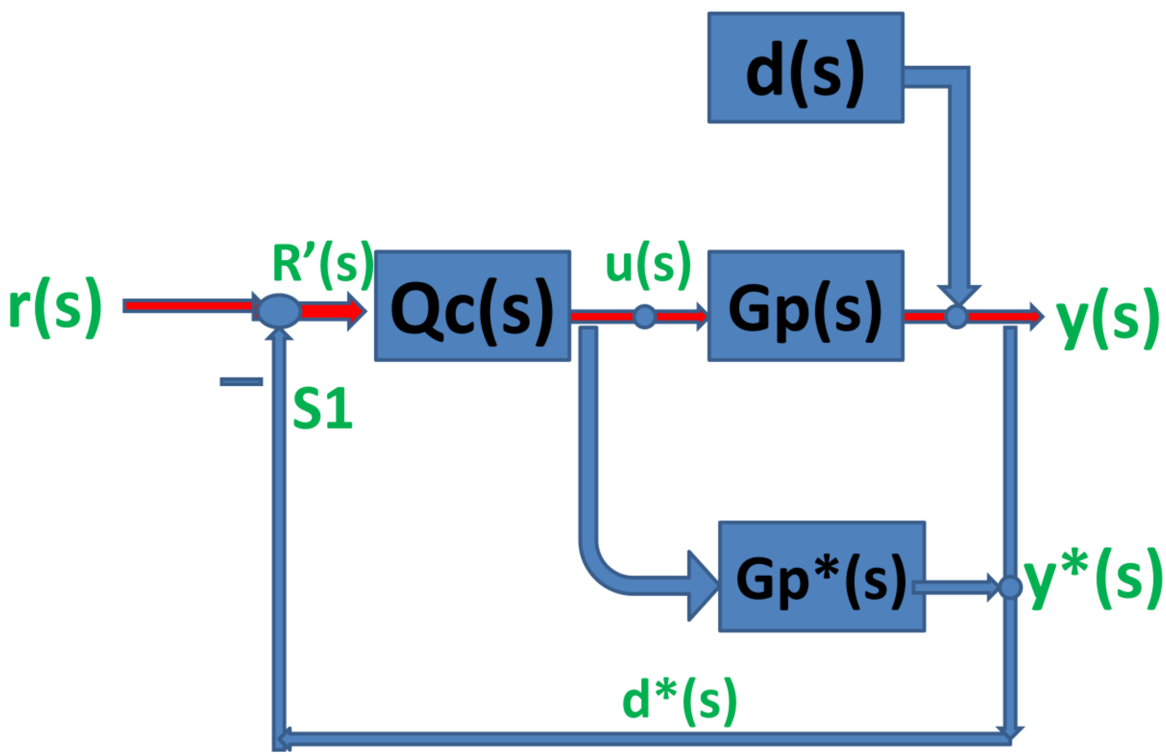



**Fig 1.3 IMC strategy**

In the above figure, d(s) is the unknown disturbance affecting the system. The manipulated input u(s) is introduced to both the process and its model. The process output, y(s), is compared with the output of the model resulting in the signal d*(s). Hence the feedback signal send to the controller is

**d*(s) = [Gp(s) – Gp*(s)].u(s) + d(s)**

In case d(s) is zero then feedback signal will depend upon the difference between the actual process and its model.

If actual process is same as process model i.e **Gp(s) = Gp*(s)** then feedback signal **d*(s)** is equal to the unknown disturbance.

So for this case **d*(s)** may be regarded as information that is missing in the model signifies and can be therefore used to improve control for the process. This is done by sending an error signal to the controller.

The error signal **R'(s)** incorporates the model mismatch and the disturbances and helps to achieve the set-point by comparing these three parameters. It is send as control signal to the controller and is given by

**R'(s) = r(s) – d*(s)** (input to the controller)

And output of the controller is the manipulated input **u(s)**. It is send to both process and its model.

$$u(s) = R'(s) \cdot Gc(s) = [r(s) - d^*(s)]\, Gc(s)$$

$$= [\, r(s) - \{[Gp(s) - Gp^*(s)].u(s) + d(s)\}\,] \cdot Gc(s)$$

$$u(s) = [\, [r(s) - d(s)]\, Gc(s)\,] / [\, 1 + \{ Gp(s) - Gp^*(s) \}\, Gc(s)\,]$$

But

$$y(s) = Gp(s) \cdot u(s) + d(s)$$

Hence, closed loop transfer function for IMC scheme is

$$\mathbf{y(s) = \{Gc(s) \cdot Gp(s) \cdot r(s) + [1 - Gc(s) \cdot Gp^*(s)] \cdot d(s)\} / \{ 1 + [Gp(s) - Gp^*(s)]\, Gc(s) \}}$$

Now if Gc(s) is equal to the inverse of the process model and if Gp(s) = Gp*(s) then perfect set point tracking and disturbance rejection can be achieved.

Also to improve the robustness of the system the effect of model mismatch should be minimized. Since mismatch between the actual process and the model usually occur at high frequency end of the systems frequency response, a low pass filter Gf(s) is usually added to attenuate the effects of process model mismatch.

**Thus the internal model controller is usually designed as the inverse of the process model in series with the low pass filter** i.e

**Gimc(s) = Gc(s). Gf(s)**

Where order of the filter is usually chosen so that the controller is proper and to prevent excessive differential control action. The resulting closed loop then becomes

**y(s) = {Gimc(s) . Gp(s) . r(s) + [1 – Gimc(s) . Gp* (s)] . d(s)} / { 1 + [Gp(s) – Gp* (s)] Gimc(s) }**

# Chapter 2

## ANALYSIS OF IMC USING SISO DESIGN TOOL

# CHAPTER 2

## 2.1 Brief introduction

SISO TOOL is a Graphical User Interface (GUI) which lets us design single-input/single-output (SISO) compensators by graphically interacting with the root locus, Bode plots of the open-loop system.  To insert the plant data into the SISO Tool, select the Import item from the File menu. By default, the control system configuration is

```
r -->[ F ]-->O--->[ C ]--->[ G ]-----+---> y
           - |                       |
             |                       |
             +--------- [H] ----------+
```

**Fig 2.1 Line diagram of a system in SISO TOOL**

where C and F are tunable compensators.

SISOTOOL (G) specifies the plant model G to be used in the SISO Tool.

SISOTOOL (G, C) and SISOTOOL (G, C, H, F) further specify values for the feedback compensator C, sensor H, and pre-filter F.

By default C, H, and F are all unit gains. Using the SISO Design Tool, we can graphically tune the gains and dynamics of the compensator (**C**) and pre-filter (**F**) using a mix of root locus and loop shaping techniques.

For example, we can use the root locus view to stabilize the feedback loop and enforce some minimum damping, and use the Bode diagrams to adjust the bandwidth, check the gain and phase margins, or add a notch filter for disturbance rejection.

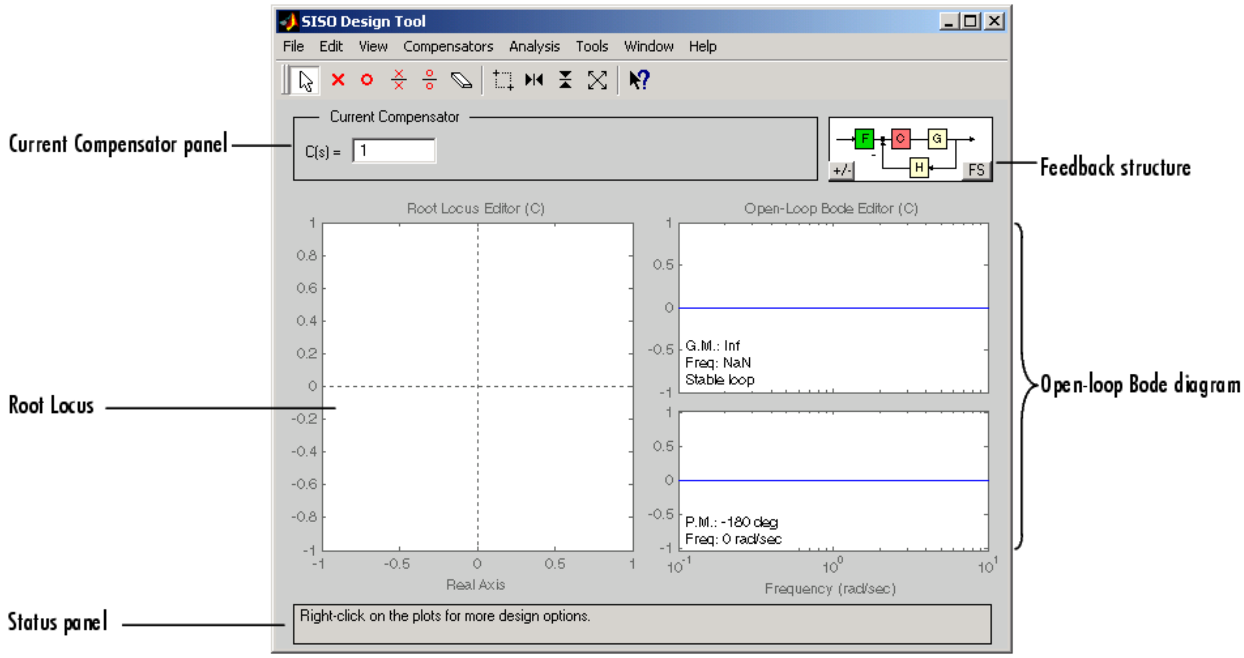



**Fig 2.2 GUI SISO design tool**

The SISO Design Tool is designed to work closely with the LTI Viewer, allowing us to rapidly iterate on your design and immediately see the results in the LTI Viewer. When we make a change in your compensator, the LTI Viewer associated with our SISO Design Tool automatically updates the response plots that we have chosen. By default, the SISO Design Tool displays the root locus and open-loop Bode diagrams for our imported systems. We can also bring up an open-loop Nichols view or pre-filter Bode diagram by selecting these items in the View menu.

Imported systems can include any of elements of the feedback structure diagram located to the right of the Current Compensator panel. We cannot change imported plant (**G**) or sensor (**H**) models, but we can use the SISO Design Tool for designing a new (or modifying an existing) pre-filter (**F**) or compensator (**C**) for your imported plant and sensor configuration.

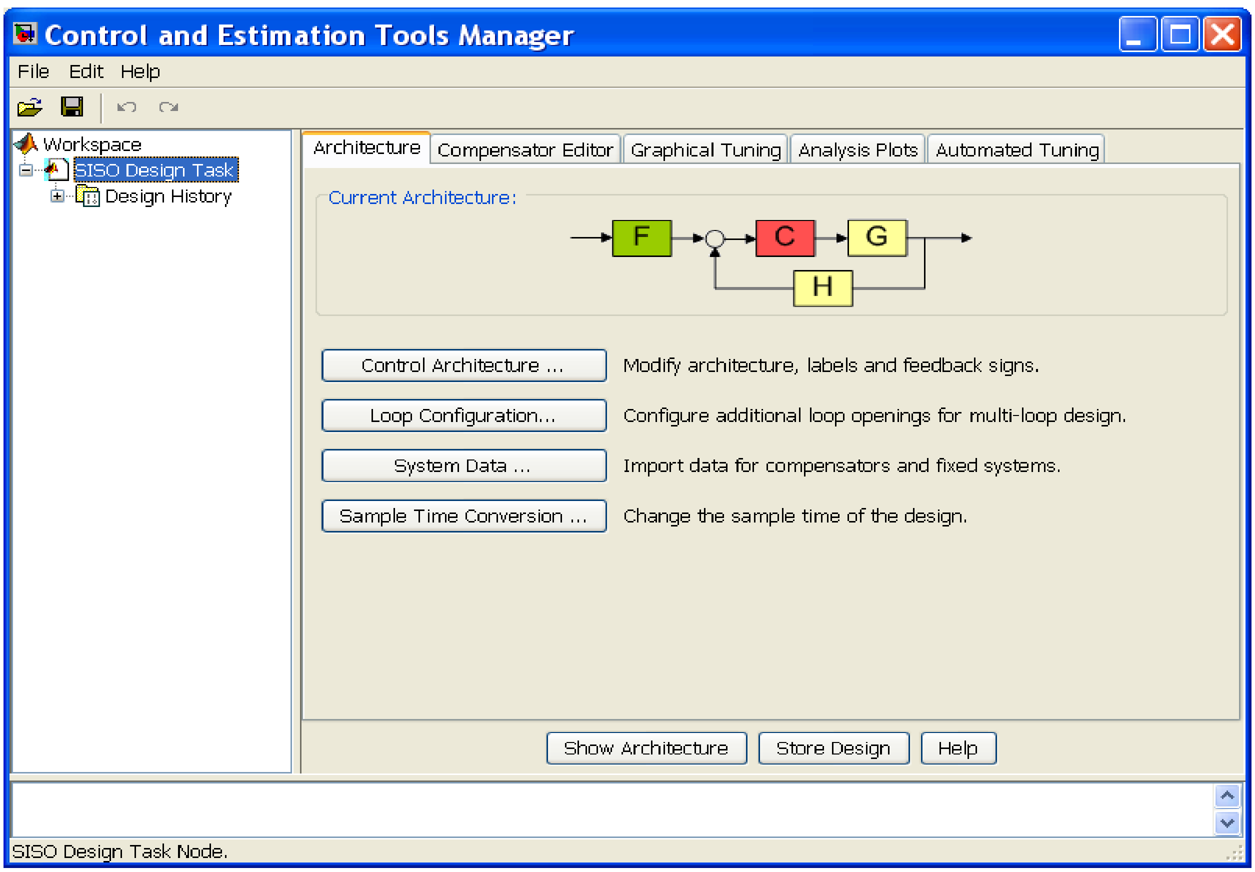
## 2.2 Using SISO TOOL for IMC implementation

IMC Design with Automatic Tuning

We will now design the compensator in an IMC structure in SISO Design Tool.

Open SISO Design Tool

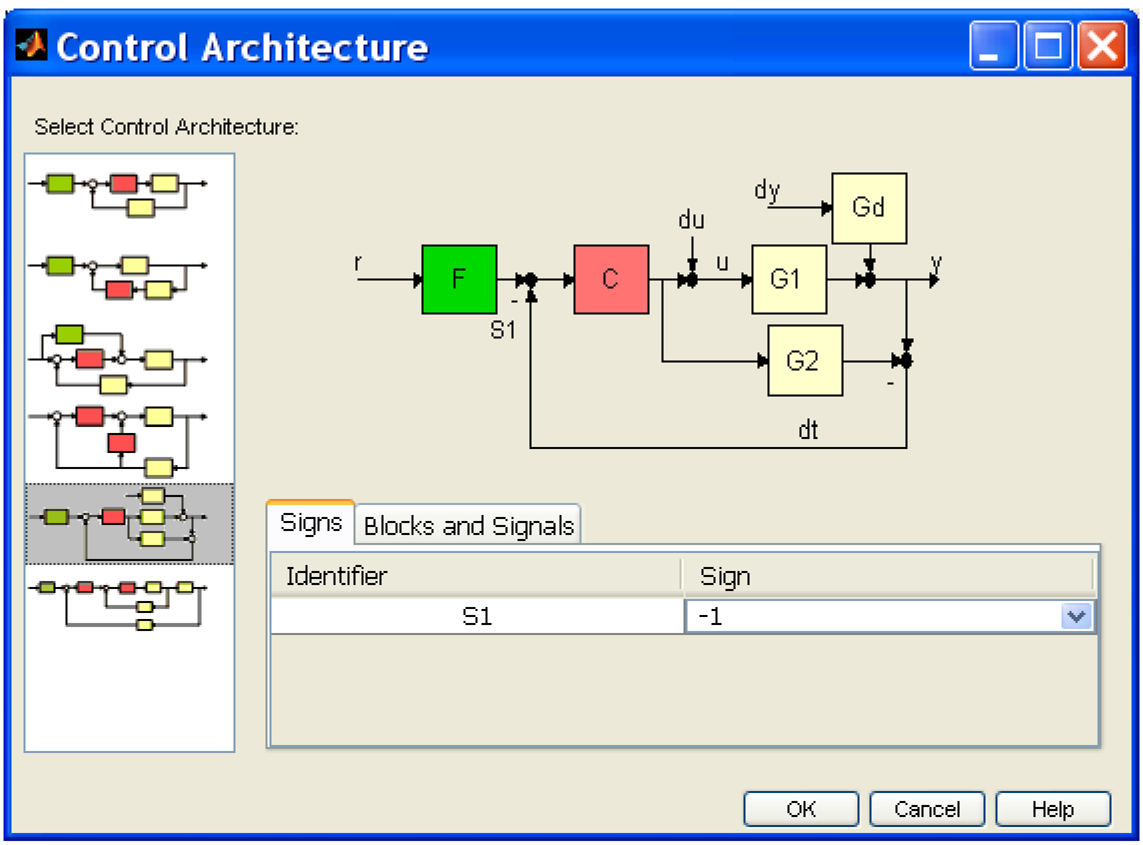
At the MATLAB® command prompt, type SISOTOOL and the Controls and Estimation Tools Manager opens.



### 2.2.1 Control architecture

- Click on the “Control Architecture” button on control tool and estimation manager.

- Select Configuration 5 for IMC structure from the panel in the Control Architecture dialog box.



### 2.2.2 Loading system data

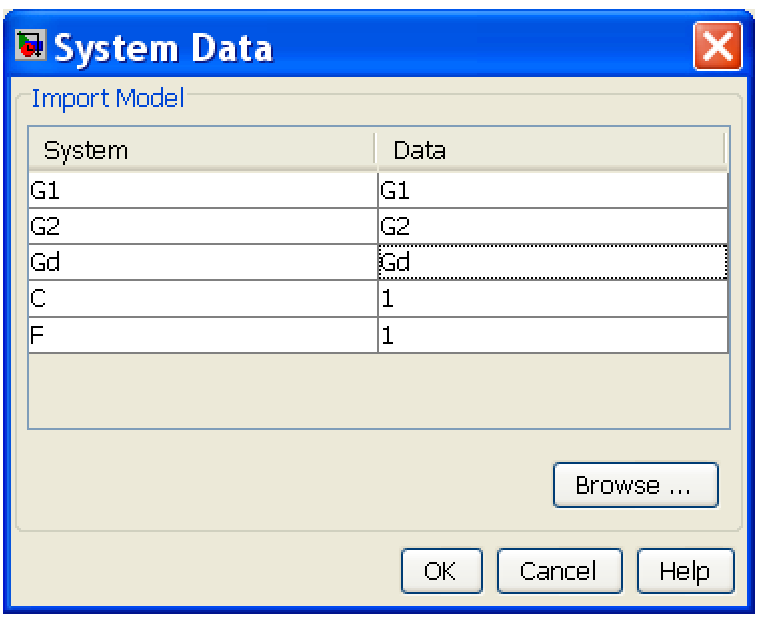
First we create the following LTI models in MATLAB command prompt:

Considering for 1st order system

```
s = tf('s');
G1 = 1 / (7 * s + 3);
G2 = G1;
Gd = 5 / (3 * s + 1);
```

Considering for 2nd order system

```
s = tf('s');
G1 = 16/ (s^2 + 2 * s + 16);
G2 = G1;
Gd = 5 / (3 * s + 1);
```



**Note**: **G1** is the real plant used; **G2** is an approximation of the real plant and it is used as the plant model in the IMC structure.

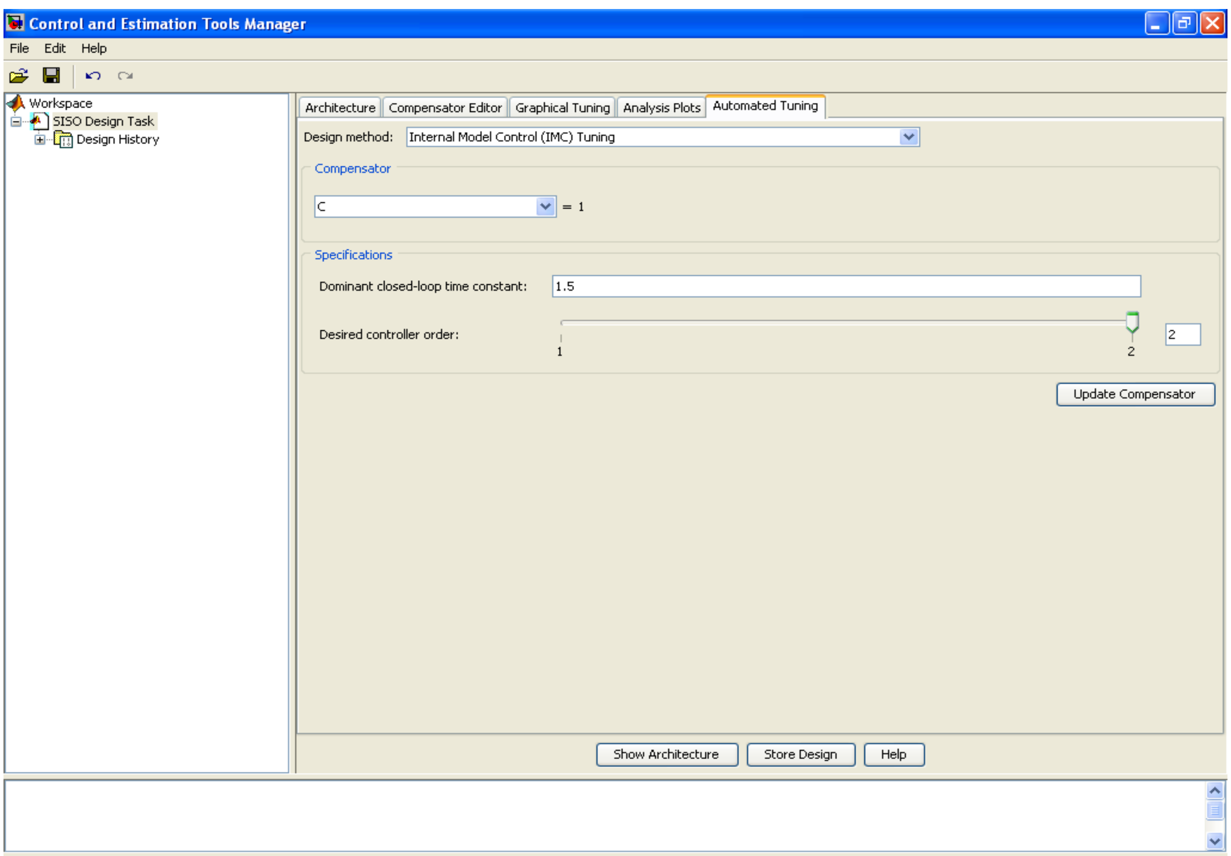
**G1 = G2** means that there is **no model mismatch**.

**Gd** is the **disturbance model**.

Now we load the system data into the Controls and Estimation Tools Manager by clicking on the System Data button. The System Data Dialog is given the above mentioned values.

### 2.2.3 Automated tuning

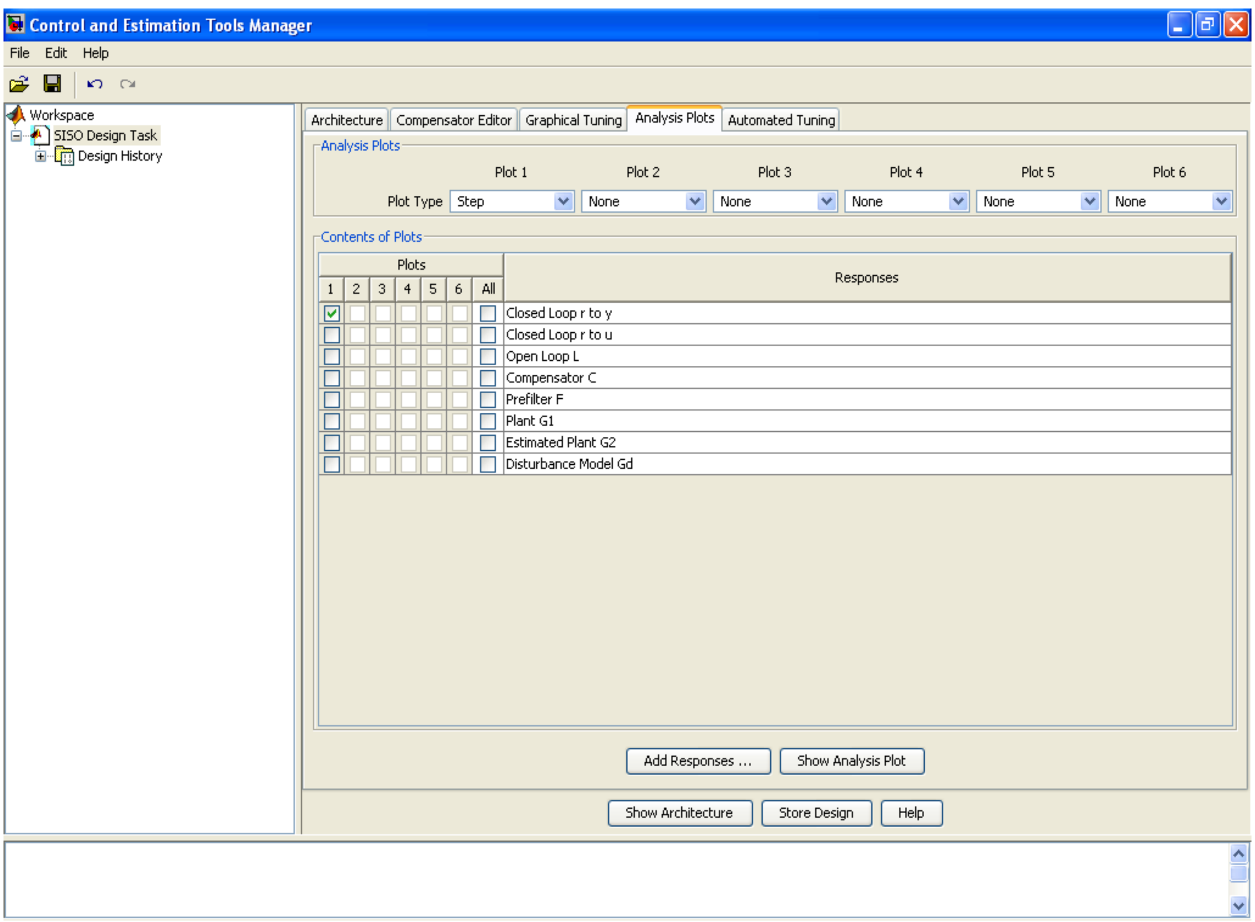
To tune the IMC compensator, we will click on the Automated Tuning on the Controls and Estimation Tools Manager and select Internal Model Control (IMC) Tuning as the design method.



Here we have taken controller of second order and now we will vary the time constant and compare different output responses for both first order and second order system. Now we take 3 different values of time constant

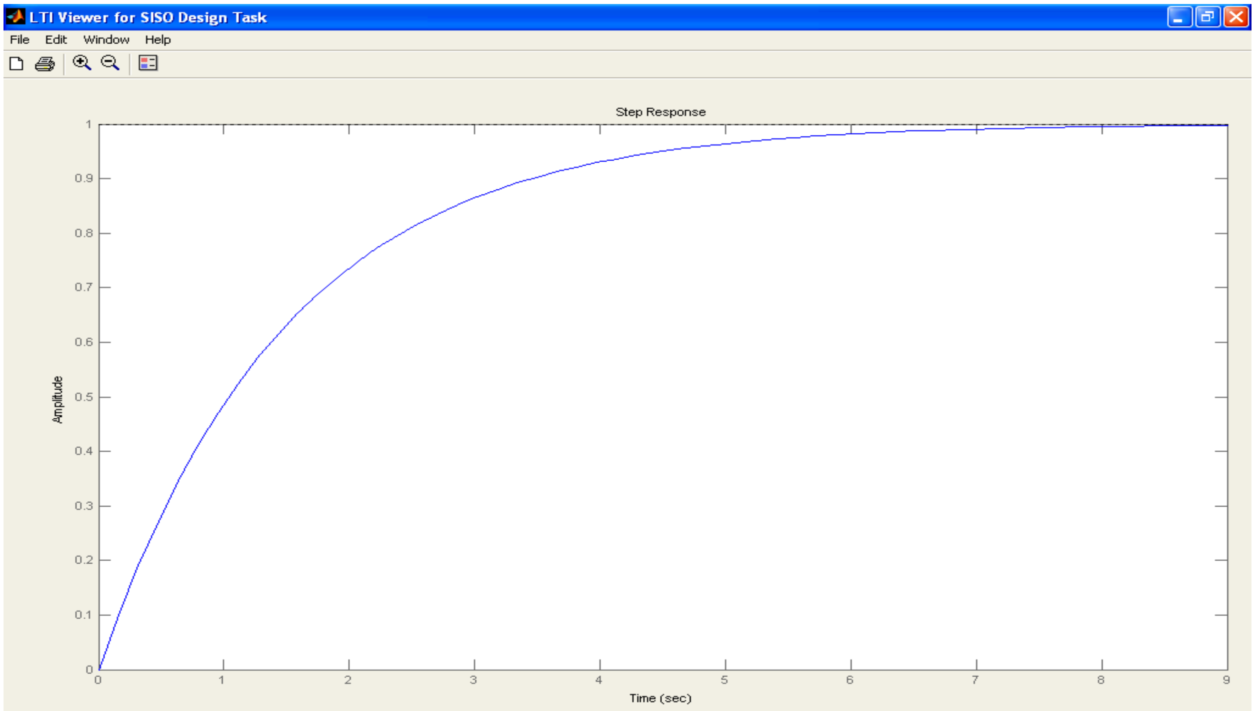
### 2.2.4 Analysis plots

To look at the closed loop response, click on the Analysis Plots on the Controls and Estimation Tools Manager, select **Step** as the plot type for Plot 1 and make Closed Loop r to y as the content of Plot 1:

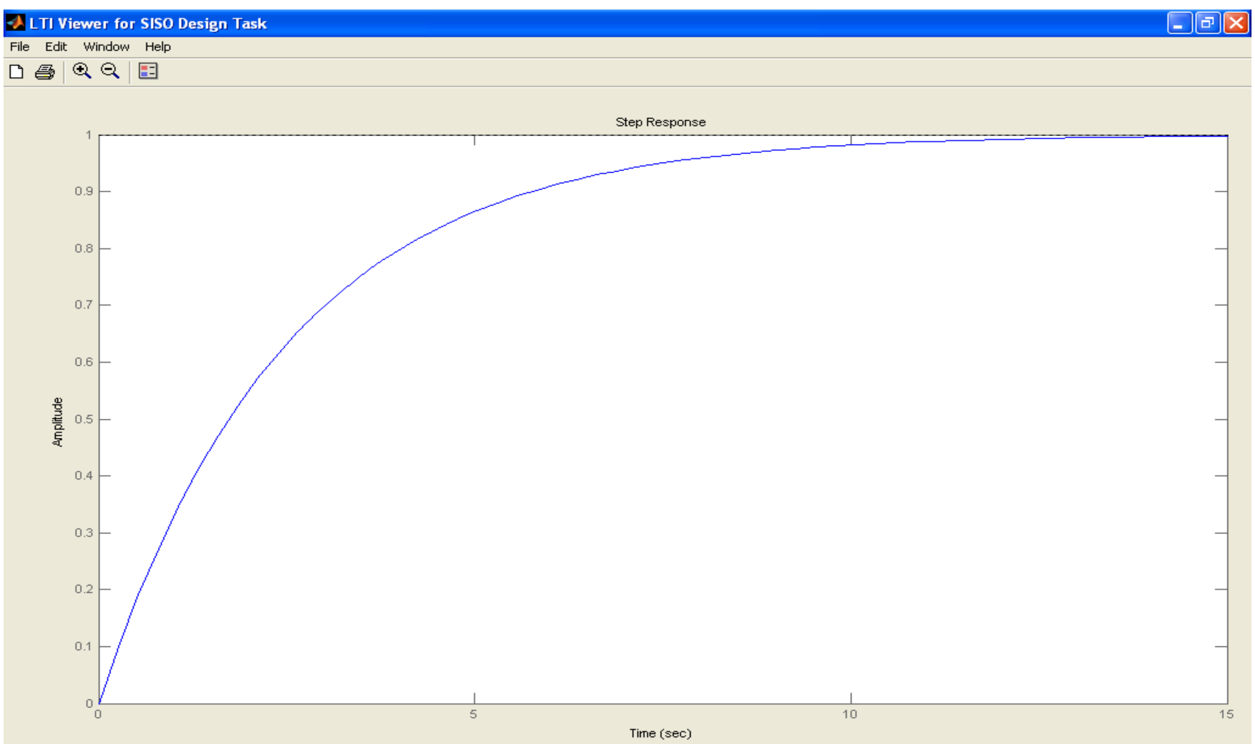


## First order plots

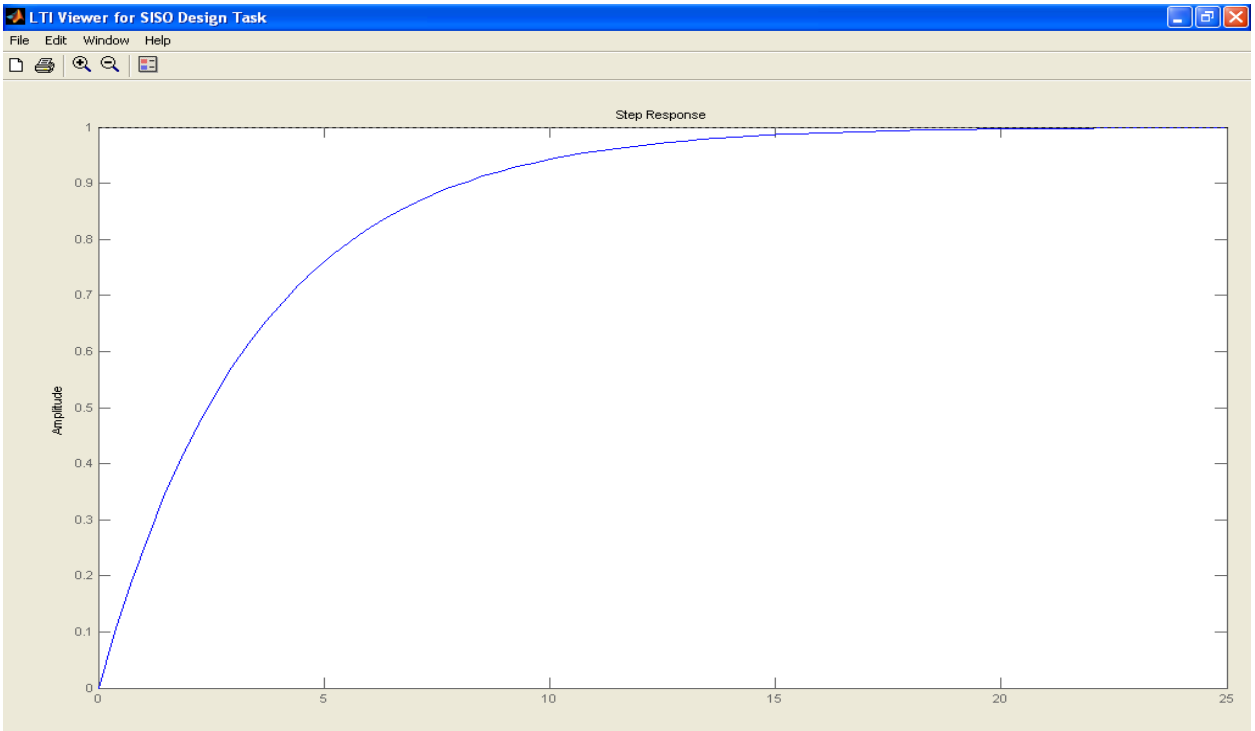
### Simulation1: For tau =1.5




### Simulation2: For tau =2.5



**Simulation3: For tau =3.5**




We have taken the first order transfer function as:

**G = 1 / (7*s + 3)**

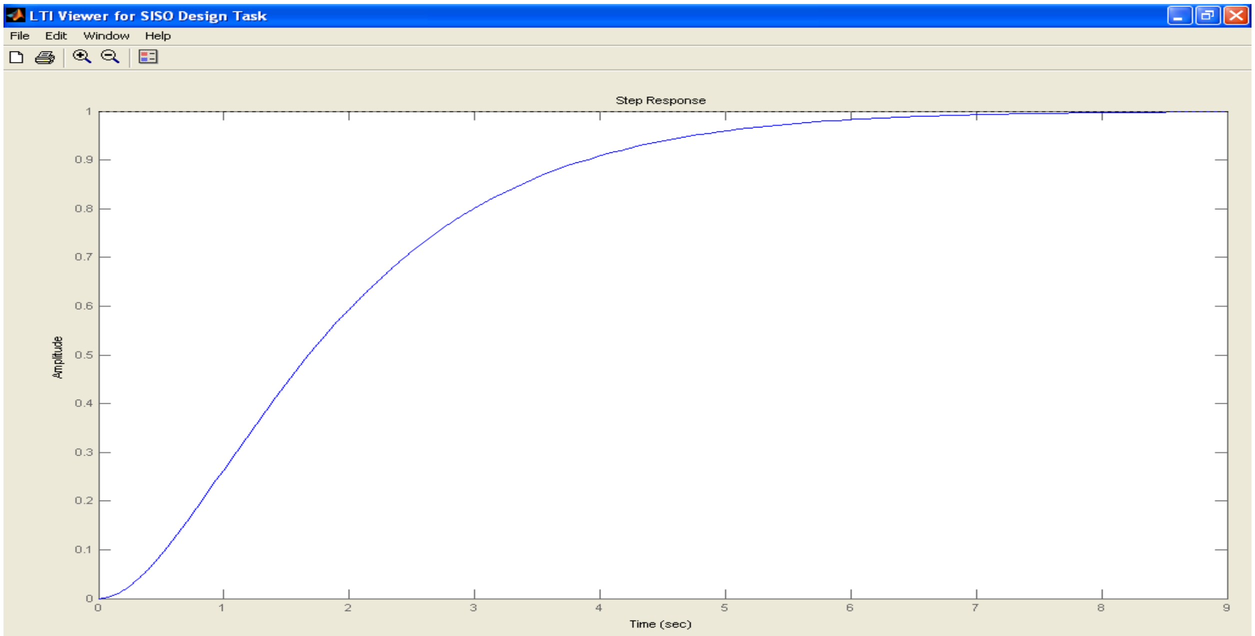
**Effect of time constant (tau) on settling time for 1st order system:**

| S. No | Values of time constant (tau) | Settling time (in sec) |
|---|---|---|
| 1 | 1.5 | 9 |
| 2 | 2.5 | 15 |
| 3 | 3.5 | 25 |

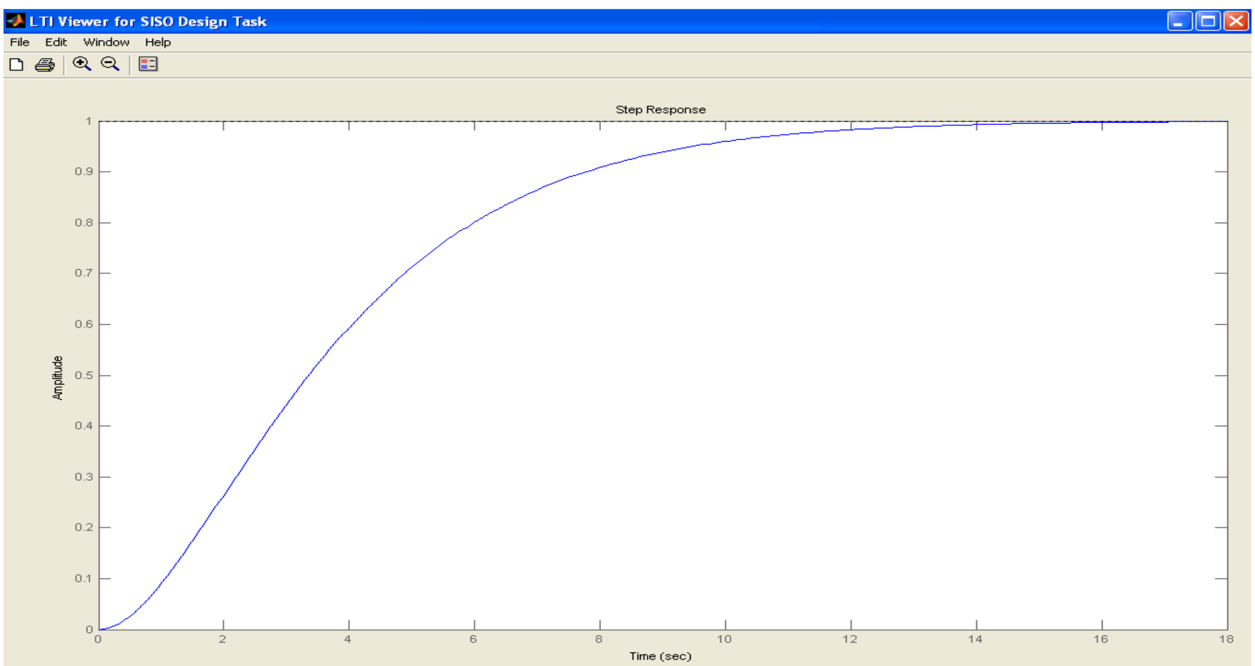
**Table 1**

## For 2nd order system

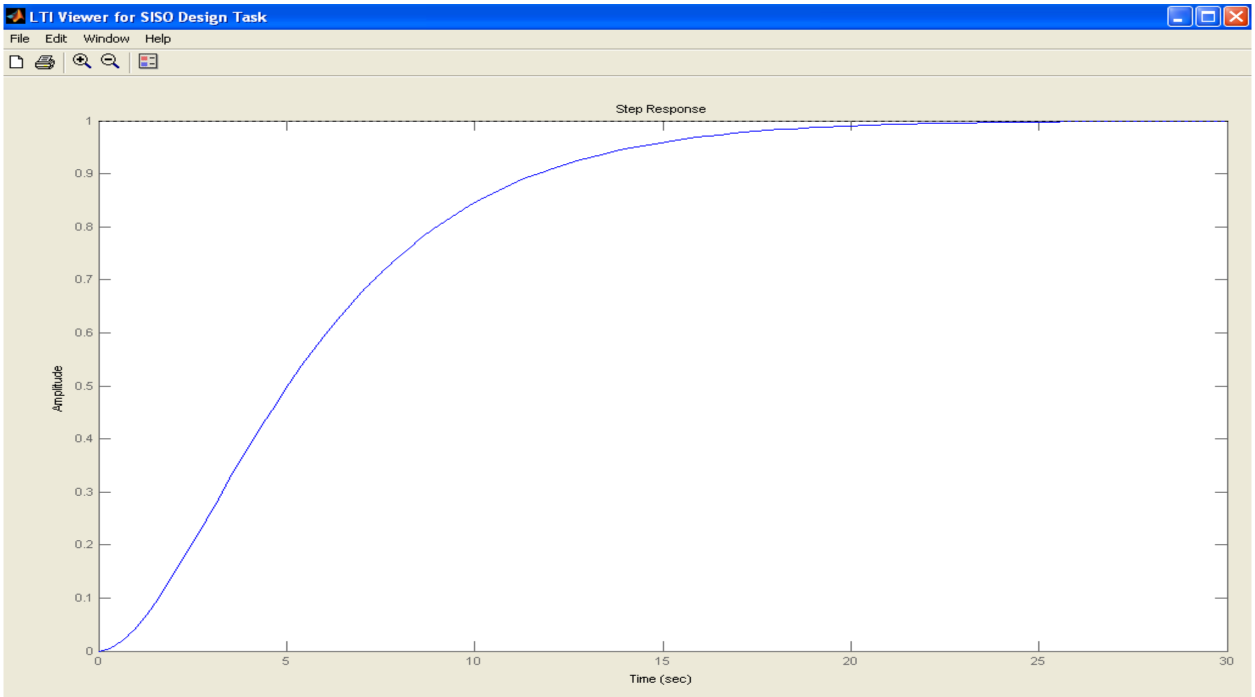
## Simulation4: For tau =1




## Simulation5: For tau =2



### Simulation6: For tau =3




We have taken the second order transfer function as:

**G = 16 / (s ^2 + 2*s +16)**

**Effect of time constant (tau) on settling time for 2nd order system:**

| S. No | Values of time constant (tau) | Settling time (in sec) |
|---|---|---|
| 1 | 1 | 8 |
| 2 | 2 | 16 |
| 3 | 3 | 25 |

**Table 2**

# Chapter 3

## IMC DESIGN PROCEDURE

### 3.1Introduction

The IMC design procedure is exactly the same as the open loop control design procedure. Unlike open loop control, the IMC structure compensates for disturbances n model uncertainties. The IMC tuning (filter) factor “lem” is used to detune for model uncertainty. It should be noted that the standard IMC design procedure is focused on set point responses but good set point responses do not guarantee good disturbance rejection, particularly for the disturbances that occur at the process inputs. A modification of the design procedure is developed to improve input disturbance rejection.

.

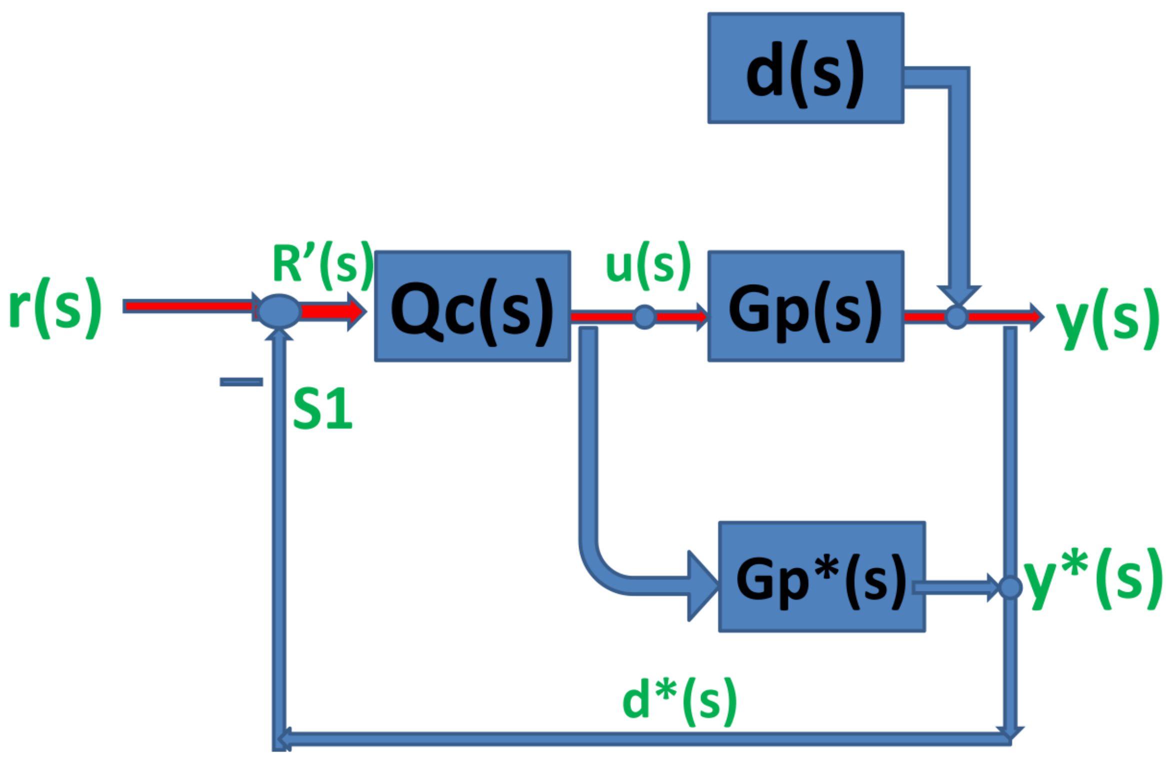



Fig 3.1 IMC design strategy

Tolerance of model uncertainty is called *robustness*. Like open loop control the disadvantage compared with standard feedback control is that IMC doesn't handle integrating or open loop unstable systems.

## 3.2 IMC design procedure

Consider a process model Gp*(s) for an actual process or plant Gp(s). The controller Qc(s) is used to control the process in which the disturbances d(s) enter into the system. The various steps in the Internal Model Control (IMC) system design procedure are:

### 3.2.1 FACTORIZATION

It means factoring a transfer function into invertible (good stuff) and non invertible (bad stuff) portions. The factor containing right hand plane (RHP) or zeros or time delays become the poles in the inverts of the process model when designing the controller. So this is non invertible portion which has to be removed from the system. Mathematically it is given as

**Gp*(s) = Gp*(+)(s)  Gp*(-)(s)**

Where

**Gp*(+)(s)** is non-invertible portion

**Gp*(-)(s)** is invertible portion

Usually we use all pass factorization

### 3.2.2 IDEAL IMC CONTROLLER

The ideal IMC controller is the inverse of the invertible portion of the process model. It is given as

**Qc*(s) = inv [ Gp*(-)(s)]**

### 3.2.3 ADDING FILTER

Now we add a filter to make our controller **proper**.

A transfer function is said to be **proper** if the order of the denominator is at least as great as the order of the numerator. If they are exactly of the same order the transfer function is said to be **semi-proper**.

If the order of the denominator is greater than the order of the numerator the transfer functions is **strictly proper.**

*Thus a controller can be physically implemented if it is proper.*

So to make the controller proper mathematically it is given as

**Qc(s) = Qc*(s) f(s) = inv [ Gp*(-)(s)] f(s)**

Where

f(s) is a low pass filter

### 3.2.4 LOW PASS FILTER f(s)

In order to improve the robustness of the system the effect of model mismatch should be minimized. Since mismatch between the actual process and the model usually occur at high frequency end of the systems frequency response, a low pass filter **f(s)** is usually added to attenuate the effects of process model mismatch.

Thus the internal model controller is usually designed as the inverse of the process model in series with the low pass filter i.e

**Qc(s) = Qc*(s) f(s) = inv [ Gp*(-)(s)] f(s)**

**Where**

**f(s)= 1/( lem* s+1) ^ n**

Where *lem* is the filter tuning parameter to vary the speed of the response of closed loop system.

Now the low pass filter can be of three types:

a) If we focus on setpoint changes, the form of filter used is

   f(s) = 1/( lem* s+1) ^ n

   here **n** is the order of the process.

b) If we focus on good tracking of ramp set point changes the filter of the form used is

   **f(s) = (n. lem. s + 1)/ (lem* s+1) ^ n**

c) If we focus on good rejection of step input load disturbances the filter of the form use is

   **f=( gamma.s+1)/( lem* s+1) ^ n**

   where gamma is any constant.

### 3.3 IMC design for 1st order system

Now we apply the above IMC design procedure for a first order system with a given process model.

- **Given process model for 1st order system : Gp*(s) = Kp*/[Tp*(s)+1]**
- Gp*(s) = Gp*(+)(s) . Gp*(-) (s) = 1 . Kp*/[Tp*(s)+1]
- Qc*(s) = inv[Gp*(-) (s) ] = [Tp*(s)+1] / Kp*
- Qc(s) = Qc*(s). f(s) = [Tp*(s)+1] / [ Kp*. (lem(s) + 1)]
- f(s) = 1 / (lem. s + 1)
- y(s) = Qc(s). Gp(s).r(s) = Gp*(+)(s) . f(s). r(s)

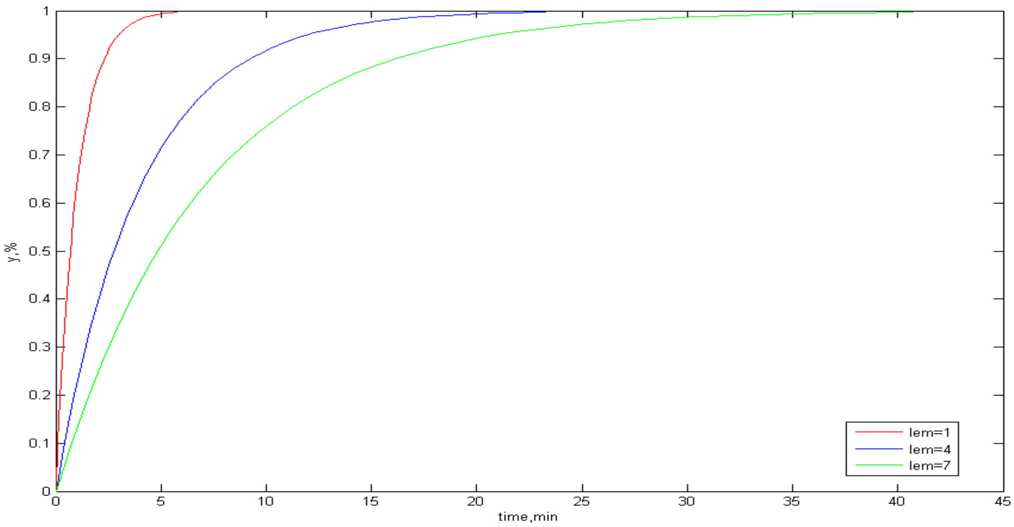
  {PERFECT MODEL}
- Output variable:

  y(s) = r(s)/(lem. s +1)
- Manipulated variable:

  u(s) = Qc(s) . r(s) = [Tp*(s)+1].r(s)/ [ Kp. (lem. s +1)

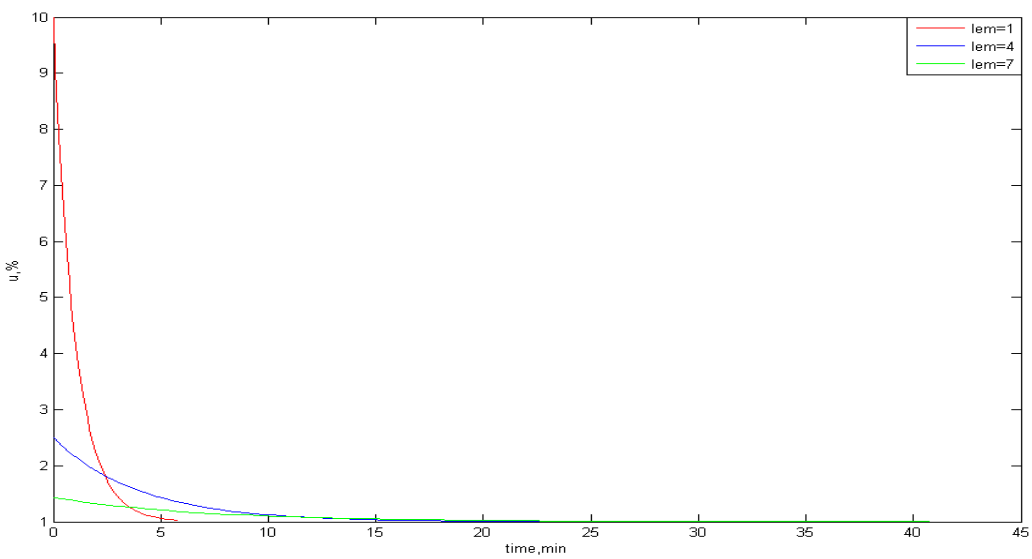
### 3.3.1Simulation plot for IMC $1^{st}$ order system

**a) Output variable response**




**Sim7: Simulation of output variable response**

**b) Manipulated variable response**




**Sim8: Simulation of manipulated variable response**

## 3.4 IMC design for $2^{st}$ order system

- **Given process model for $2^{nd}$ order system: Gp*(s) = [-9s + 1] / [ (15s +1) (3s + 1)]**
- Gp*(s) = Gp*(+)(s) . Gp*(-) (s) = (-9s + 1 )/(9s + 1) . [9s + 1] / [ (15s +1) (3s + 1)]

- Qc*(s) = inv[Gp*(-) (s) ] = [ (15s +1) (3s + 1)] / ( 9s + 1)

- Qc(s) = Qc*(s). f(s) = [ (15s +1) (3s + 1) / ( 9s + 1) ] . [ 1 / (lem . s + 1) ]

- f(s) = 1 / (lem . s + 1)

- y(s) = Qc(s). Gp(s).r(s) = Gp*(+)(s) . f(s). r(s)

  {PERFECT MODEL}

- Output variable:

  y(s) = [-9s + 1] / [ (15s +1) (3s + 1)] . r(s)

  = [-9s+1] / [ 9 lem s^2 + (9 +lem ) s +1]

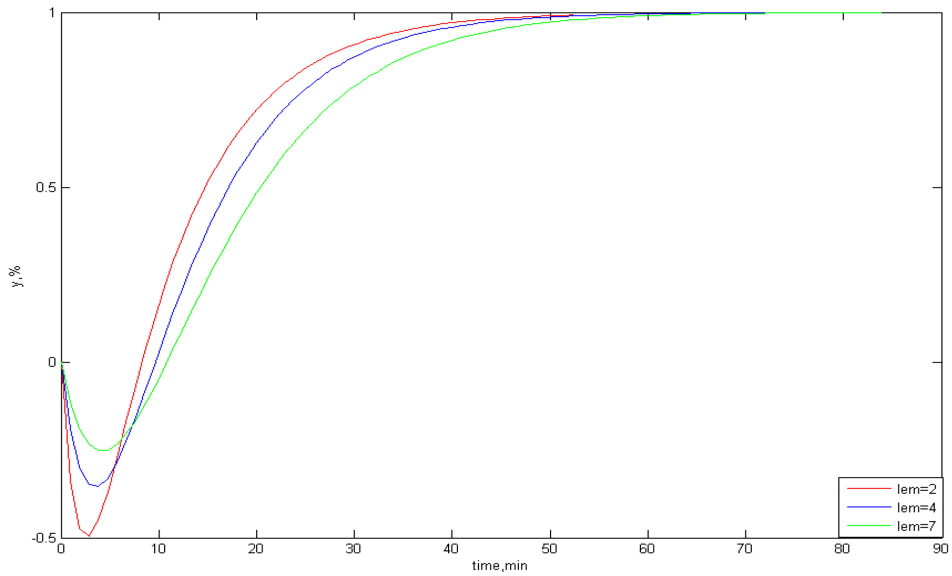
- Manipulated variable:

  u(s) = Qc(s) . r(s) = [ (15s +1) (3s + 1) / ( 9s + 1)(lem . s +1) ] . r(s)

  = [(45 s^2 + 18 s +1)/ ( 9 lem s^2 + (9 +lem ) s +1)] . r(s)

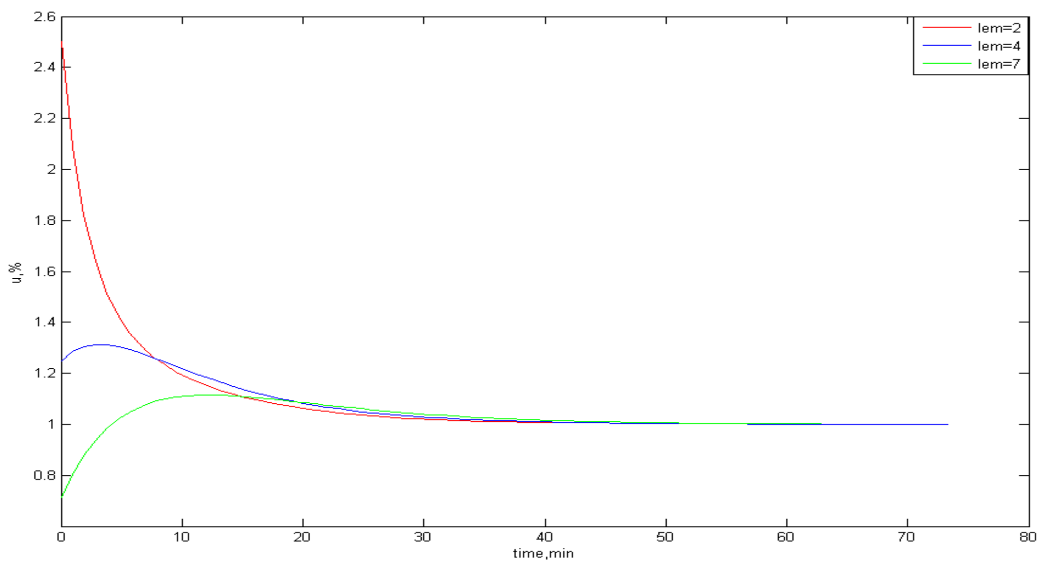
### 3.4.1 Simulation plot for IMC 2st order system

**a) Output variable response**




**Sim9: Simulation of output variable response**

**b) Manipulated variable response**




**Sim10: Simulation of manipulated variable response**

# Chapter 4

## IMC BASED PID

## CHAPTER 4

### 4.1Introduction

The IMC structure can be rearranged to form a standard feedback control system that can easily handle open loop unstable system as not the case with IMC. This modification of the IMC design procedure is developed to improve the input disturbance rejection. The IMC based PID structure which uses a standard feedback structure uses the process model in an implicit manner i.e. PID tuning parameters are often adjusted based on the transfer function model but it is not always clear how the process model affects the tuning decision. In the IMC procedure the controller Qc(s) is directly based on the good part of the process transfer function. Also the IMC formulation generally results in only one tuning parameter, the close loop time constant (filter tuning factor). The IMC based PID tuning parameters are then the function of this time constant. The selection of the closed loop time constant is directly related to the robustness (sensitivity to the modular of the closed loop system). Also, for open loop unstable processes it is necessary ti implement the IMC strategy in standard feedback form, because the IMC suffers from internal stability problems. Though the IMC based PID controller will not give the same performance when there are process time delays because the IMC based PID procedures uses an approximation for the dead time. But if the process has no time delays and the inputs do not hit a constraint then the IMC based PID controller give the same performance as does the IMC.

### 4.2 IMC based PID structure

In the IMC structure the point of comparison between the process and the model output can be moved as shown in the figure below to form a standard feedback structure which is nothing but another equivalent feedback form of IMC structure know n as IMC based PID structure.

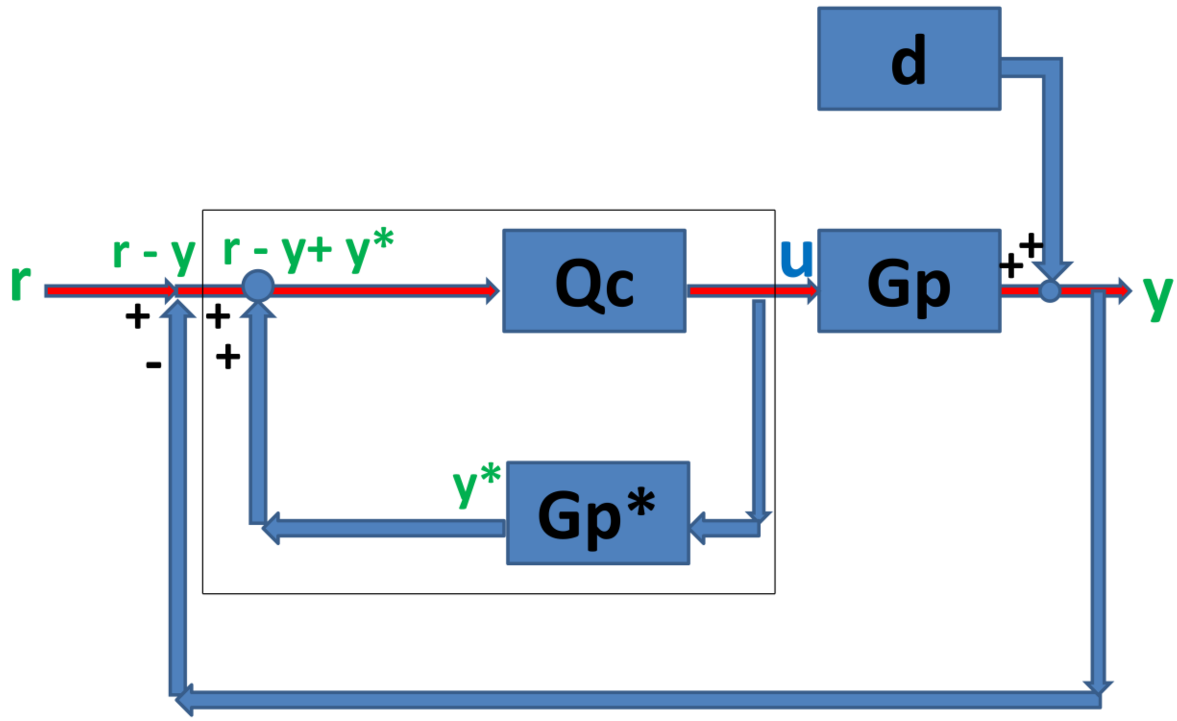



Fig 4.1 IMC based PID design

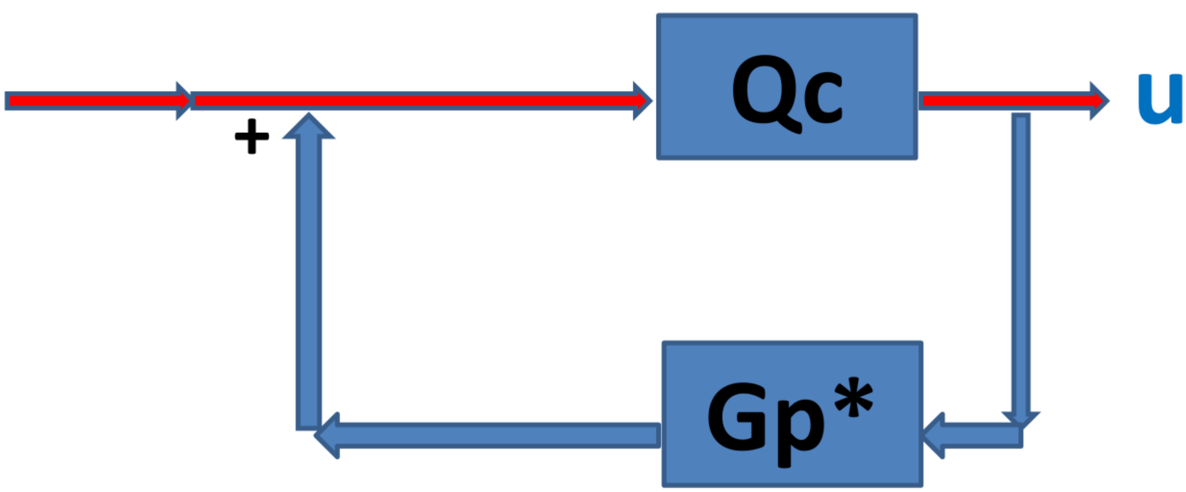



Fig 4.2 inner loop of rearranged IMC structure

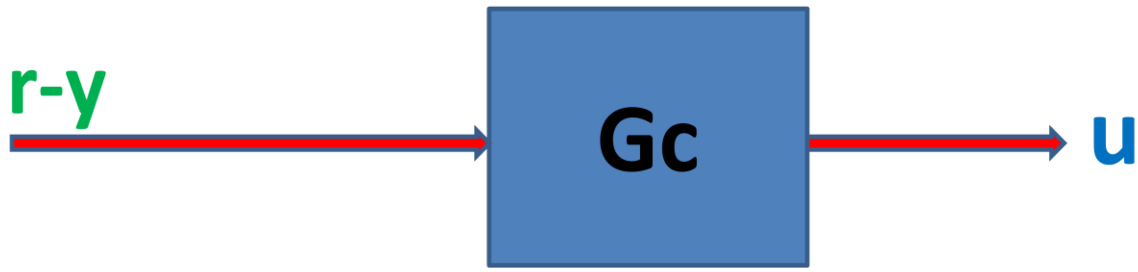



Fig 4.3 equivalent IMC rearranged structure

### 4.3 IMC based PID design procedure

Consider a process model Gp*(s) for an actual process or plant Gp(s). The controller Qc(s) is used to control the process in which the disturbances d(s) enter into the system. The various steps in the Internal Model Control (IMC) system design procedure are:

#### 4.3.1 FACTORIZATION

It means factoring a transfer function into invertible (good stuff) and non invertible (bad stuff) portions. The factor containing right hand plane (RHP) or zeros or time delays become the poles in the inverts of the process model when designing the controller. So this is non invertible portion which has to be removed from the system. Mathematically it is given as

**Gp*(s) = Gp*(+)(s)  Gp*(-)(s)**

Where

**Gp*(+)(s)** is non-invertible portion

**Gp*(-)(s)** is invertible portion

Usually we use all pass factorization

### 4.3.2 IDEAL IMC CONTROLLER

The ideal IMC controller is the inverse of the invertible portion of the process model. It is given as

**Qc*(s) = inv [ Gp*(-)(s)]**

### 4.3.3 ADDING FILTER

Now we add a filter to make our controller **proper**.

A transfer function is said to be **proper** if the order of the denominator is at least as great as the order of the numerator. If they are exactly of the same order the transfer function is said to be **semi-proper**.

If the order of the denominator is greater than the order of the numerator the transfer functions is **strictly proper.**

*Thus a controller can be physically implemented if it is proper*.

So to make the controller proper mathematically it is given as

**Qc(s) = Qc*(s) f(s) = inv [ Gp*(-)(s)] f(s)**

where f(s) is a low pass filter

### 4.3.4 LOW PASS FILTER [f(s)]

In order to improve the robustness of the system the effect of model mismatch should be minimized. Since mismatch between the actual process and the model usually occur at high frequency end of the systems frequency response, a low pass filter **f(s)** is usually added to attenuate the effects of process model mismatch.

Thus the internal model controller is usually designed as the inverse of the process model in series with the low pass filter i.e

**Qc(s) = Qc*(s) f(s) = inv [ Gp*(-)(s)] f(s)**

**Where**

**f(s) = 1/( lem* s+1) ^ n**

Where *lem* is the filter tuning parameter to vary the speed of the response of closed loop system.

Now the low pass filter can be of three types:

a) If we focus on setpoint changes, the form of filter used is

f(s) = 1/( lem* s+1) ^ n

here **n** is the order of the process.

b) If we focus on good tracking of ramp set point changes the filter of the form used is

**f(s) = (n. lem. s + 1)/ (lem* s+1) ^ n**

c) If we focus on good rejection of step input load disturbances the filter of the form use is

**f = ( gamma.s+1)/( lem* s+1) ^ n**

where gamma is any constant.

### 4.3.5 Equivalent standard feedback controller

By rearranging the IMC we obtain equivalent standard feedback controller using transformation.

**Gc=Qc/(1-Qc Gp*)**

We write this expression in the form of a ratio between two polynomials.

### 4.3.6 Comparison with standard PID controller

Now we compare with PID Controller transfer function

For first order :

Gc(s) = [Kc . (Ti .s + 1)]/ (Ti . s)

And find Kc and Ti ( PI tuning parameters)

Similarly for $2^{nd}$ order we compare with the standard PID controller transfer function given by :

Gc(s) = Kc . [(Ti . Td . s^2+Ti . s+1)/Ti . s]. [ 1/ Tf . s+1]

Where

T = Tau (any constant)

Ti = integral time constant

Td = derivative time constant

Tf = filter tuning factor

Kc = controller gain

Now we perform closed loop simulations for above procedure and adjust lem (lemda) considering a trade off between performance and robustness (sensitivity to model error).

## 4.4 IMC based PID for $1^{st}$ order system

Now we apply the above IMC based PID design procedure for a first order system with a given process model.

- **Given process model : Gp*(s) = Kp*/[Tp*(s)+1]**

- Gp*(s) = Gp*(+)(s) . Gp*(-) (s) = 1. Kp*/[Tp*(s)+1]

- Qc*(s) = inv[Gp*(-) (s) ] = [Tp*(s)+1] / Kp*

- Qc(s) = Qc*(s). f(s) = [Tp*(s)+1] / [ Kp*. (lem(s) + 1)]
- f(s) = 1 / (lem. s + 1)

- Equivalent feedback controller using transformation

  Gc(s) = Qc (s)/(1-Qc(s) Gp*(s)) = [{Tp*(s)+1} / { Kp*. (lem(s) + 1)}]/

  [{1- Kp*/ (Tp*(s) +1)}. {Tp*(s) +1} / { Kp*. (lem(s) + 1)}]

- Gc(s) = {Tp(s)+1} / Kp . lem. s (it is standard feedback controller for IMC )

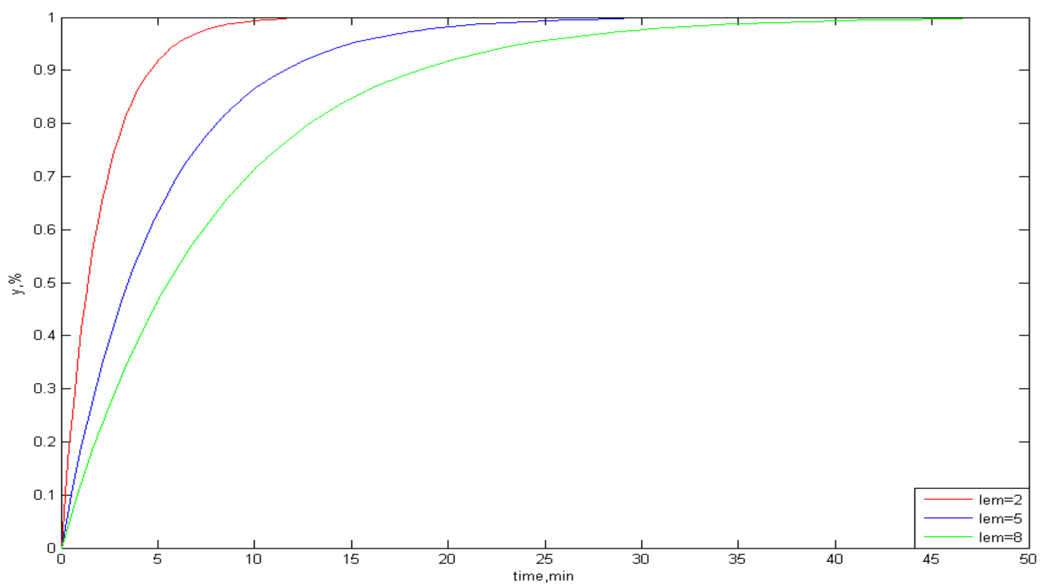
- Gc(s) = [Kc . (Ti .s + 1)]/ (Ti . s) (transfer function for PI controller)

- PI tuning parameters

  Kc = Tp / Kp. lem

  Ti = Tp

### 4.4.1 Simulation for IMC based PID $1^{st}$ order system




**Sim11: Simulation of IMC based PID $1^{st}$ order system**

## 4.5 IMC based PID for 2nd order system

Now we apply the above IMC based PID design procedure for a second order system with a given process model.

- **Given process model : Gp*(s) = Kp*/[(Tp1*(s)+1).(Tp2*(s)+1)]**

- Gp*(s) = Gp*(+)(s) . Gp*(-) (s) = 1 . Kp*/[Tp*(s)+1]

- Qc*(s) = inv[Gp*(-) (s) ] = [Tp*(s)+1] / Kp*

- Qc(s) = Qc*(s). f(s) = [Tp*(s)+1] / [ Kp*. (lem(s) + 1)]

- f(s) = 1 / (lem. s + 1)

- Equivalent feedback controller using transformation

  Gc(s) = Qc (s)/(1-Qc(s) Gp*(s))

  =[Tp1 . Tp2 s^2 +( Tp1 +Tp2)s+1] / [Kp . lem . s]

  (it is the transfer function for the equivalent standard feedback controller )
- Gc(s) = [Kc . (Ti .Td.s^2 + Ti.s+1)]/ [Ti . s] (transfer function for ideal PID controller for second order)

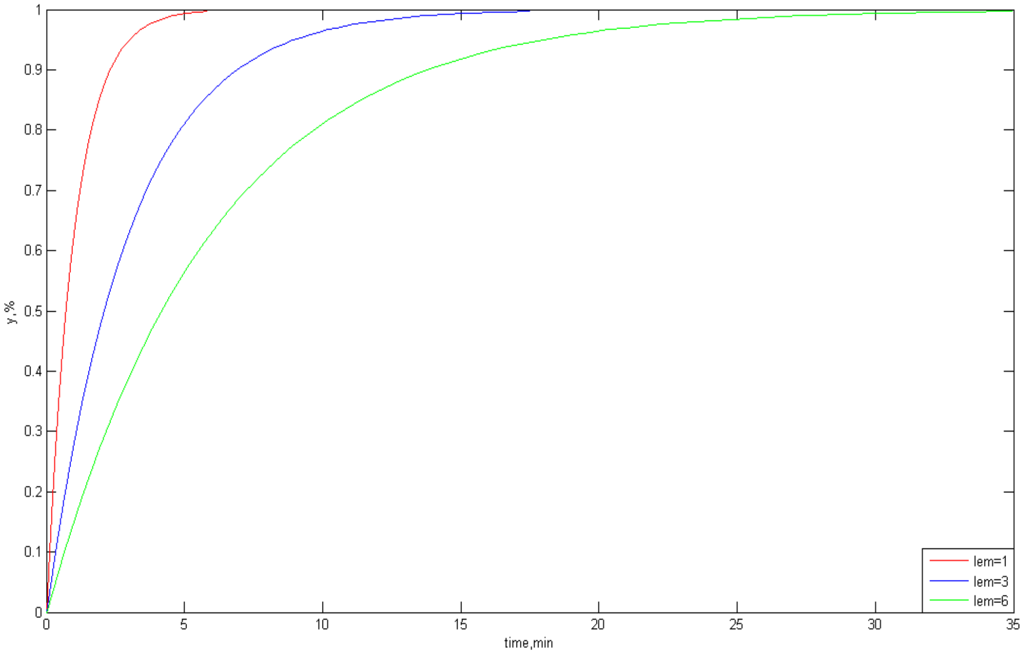
- PID tuning parameters (on comparison)

  Kc = (Tp1 + Tp2) / (Kp. lem)

  Ti = Tp1 + Tp2

  Td=Tp1 . Tp2/ (Tp1 + Tp2)

### 4.4.1 Simulation for IMC based PID 2nd order system




**Sim12: Simulation of IMC based PID 2nd order system**

# APPLICATIONS

- **At steady state, the controller will give offset free response (perfect control at steady state).**

- **The controller can be used to shape both the input tracking and disturbance rejection responses.**

- **Provides time delay compensation.**

# CONCLUSION & FUTURE WORKS

The study of Internal Model Control (IMC) and its applications for design of compensator used in IMC Model shows that our controller used is fairly robust towards uncertainty in plant parameters and it can be successfully implemented to any industrial process**.** At the same time, for practical applications or an actual process in industries IMC based PID controller algorithm is simple and robust to handle the model inaccuracies and hence using IMC-PID tuning method a clear trade-off between closed-loop performance and robustness to model inaccuracies is achieved with a single tuning parameter.

Based on the concept of design of model of the actual process, the IMC design procedure can help to solve many critical problems at the industrial level. It also provides a good solution to the process with significant time delays which is actually the case with working in real time environment. As far as the tuning of the controller is concerned we have an optimum filter tuning factor $\lambda$ (lambda) value which compromises the effects of discrepancies entering into the system to achieve the best performance. Thus, what we mean by the best filter structure is the filter that gives the best PID performance for the optimum $\lambda$ value.

We have also shown that the IMC procedure can be used to design the PID-type feedback controllers. If the process has no time delays the IMC based PID controllers will perform same as the IMC .If the process has the RHP zero then the specified closed loop response must also have a RHP zero. The IMC based PID procedure provides a clear method for handling this. Also the standard IMC filter results in good set point response performances. Although the IMC design procedure is identical to the open loop control design procedure, the implementation of IMC results in a feedback system. Thus, IMC is able to compensate for disturbances and model uncertainty while open loop control is not. Also IMC must be detuned to assure stability if there is model uncertainty.

# REFERENCES


- Process Control, Modeling Design & Simulation by B. Warne Bequette.

- Industrial Automation & process control by Jon Stenerson

- Process dynamics and control By Dale E Seborg, Mellichamp and Edgar

- M. Shamsuzzoha, Moonyong Lee* Jietae Lee1-
  School of Chemical Engineering and Technology, Yeungnam University

- Chemical and Process Engineering
  UNIVERSITY OF NEWCASTLE UPON TYNE

- Morari, M.,and E. Zafiriou, Robust process control(1989)

- “Improved filter design in internal model control” by C.J.Gombas, R.D.Braatz (1996)


## References from internet:


- http://www.pdfqueen.com
- http://www.sciencedirect.com/
- http://books.google.co.in/
- http://www.mathworks.com/

- www.topedu**sites**.com/
- www3.interscience.wiley.com
- journals.cambridge.org
- http:// cheer.sdsu.edu
- lorien.ncl.ac.uk
- http://ftp.math.hkbu.edu.hk
- http://yeungnam.academia.edu/nuvan/Papers
- http://www.waljatcolleges.edu.om/Downloads/PDFs/Project%20Format.doc
- http://ieeexplore.ieee.org/xpls/abs_all.jsp?isnumber=29279&arnumber=1322132&count=19&index=13
- http://doi.ieeecomputersociety.org/10.1109/ISDA.2006.30
- http://www.swarthmore.edu/NatSci/echeeve1/Ref/LPSA/PartialFraction/PartialFraction.html
- http://www.swarthmore.edu/NatSci/echeeve1/Ref/LPSA/PartialFraction/PartialFraction.html
- http://ecce6.kt.dtu.dk/cm/content/abstract/363/